
Quality Criteria of Genetic Algorithms for Construction of Phylogenetic Trees

T. H. REIJMERS, R. WEHRENS, L. M. C. BUYDENS

Laboratory for Analytical Chemistry, University of Nijmegen, 6525 ED Nijmegen, The Netherlands

Received 27 November 1998; accepted 2 February 1999

ABSTRACT: In this article the suitability of two different tree representations for the construction of phylogenetic trees with genetic algorithms is examined. On the one hand tree topologies are represented by means of a distance matrix while on the other hand the Prüfer number tree representation is used. To assess the adequacy of both approaches a set of recently proposed quality criteria is used. The quality criteria can be used to monitor genetic algorithm approaches differing in configuration and setup, fitness function, or representation. In addition to the criteria for the repeatability of the optimization, criteria for the coverage of the search space are also used. On the basis of the optimization results of simulated data, the quality criteria show, in contrast to the error plots, a clear difference in the efficiency of both representations. It is concluded that the Prüfer number representation yields a diverse set of good quality topologies while the distance matrix representation mainly returns the optimal topology. © 1999 John Wiley & Sons, Inc. *J Comput Chem* 20: 867–876, 1999

Keywords: phylogenetics; genetic algorithms; quality criteria; optimization; bifurcating trees

Introduction

Many functional proteins are believed to have descended from a common ancestor. In the course of time the ancestor mutated, resulting in

Correspondence to: Dr. L. M. C. Buydens; e-mail: lbuydens@sci.kun.nl

Contract/grant sponsors: Dutch Foundation for Chemical research (SON); Dutch Organisation for Scientific Research (NWO)

the proteins known today. The divergence over evolutionary history can be described with a tree-like structure termed a phylogeny.¹ To construct phylogenetic trees, similarity data can be used, which is obtained from pairwise comparisons of the DNA or protein sequences. In phylogenetic analysis, a large amount of time is spent evaluating different tree topologies in an effort to find the tree that meets a criterion of optimality given the data. As the number of descendants increases, the number of possible tree topologies increases very quickly. The total number of possible topologies

(#Top) of a bifurcating tree of x descendants is

$$\#Top = \prod_{i=3}^x (2i - 5). \quad (1)$$

From a certain number of descendants it is no longer feasible to perform an exhaustive search for the optimal topology. Therefore, a lot of phylogenetic methods use simple clustering algorithms to obtain a starting topology that will be further optimized by means of local optimization methods. However, the optimal tree topology will only be found if the starting topology supplied by the clustering algorithm is "good enough." If this is not the case, the local rearrangements do not result in the desired topology. To find the optimal tree topology without testing each topology separately, global optimization techniques can be used.

Global optimization techniques have proven to be successful in many problems where the optimal solution has to be found out of many possible solutions, although there is no guarantee of finding it. One of the more frequently used techniques is the genetic algorithm.^{2,3} The reason that genetic algorithms are so popular is their versatility. The large number of parameters makes the algorithm versatile so that it can be tailored to the requirements and characteristics of a specific application.⁴ On the other hand, because there are so many parameters that can be altered, it often takes more time to optimally and correctly configure the algorithm than to obtain the solution.

An additional advantage of the usage of genetic algorithms is that they yield a set of good quality solutions. This is especially important for the construction of phylogenetic trees. Because of the incompleteness and the errors in the obtained distance data, it is better to have a set of good quality topologies to choose from than to have only one topology.

Usually, the performance of a certain genetic algorithm approach is determined solely on the basis of the quality of its best solution (the final fitness value). Recently, a set of quality criteria were proposed to evaluate the total performance of the algorithm.⁵ The criteria can be used to facilitate the finding of parameter values that result in an optimal performance of the algorithm. Because no direct use is made of fitness values, other factors that influence the performance of the algorithm, like the fitness function or representation,^{6,7} can also be monitored using those criteria. This article illustrates how the influence of two different representations on the performance of the

genetic algorithm for the construction of phylogenetic trees can be visualized.

In the next section, the different tree representations and the quality criteria are thoroughly explained.

Methods

GENETIC ALGORITHM APPROACH

The construction of phylogenetic trees with genetic algorithms was previously described.⁸⁻¹⁰ In all approaches, an initial population of tree topologies is generated where each topology is represented as a string of numbers. After evaluating all strings, a new population is generated by means of selection, recombination, and mutation operators. The whole process is repeated until, driven by the principle of survival of the fittest, the optimal tree topology is found. The main difference between the approaches mentioned above is the usage of the recombination operator in the algorithm. Matsuda⁸ and Lewis⁹ both use recombination operators that are specially designed for the construction of phylogenetic trees; Reijmers et al.¹⁰ use standard recombination operators. To still be able to optimize tree topologies, ref. 10 uses a tree representation that does not result in forbidden trees after applying a standard crossover.

TOPOLOGY REPRESENTATION I: DISTANCE MATRIX

For a given distance matrix, the topology of the tree is fixed after applying a clustering method. By changing the distance matrix and applying the same clustering method, the topology of the tree can be changed. Accordingly, one way to optimize different tree topologies is to let the genetic algorithm adapt the distance matrix that after application of a clustering algorithm results in different tree topologies (see Fig. 1). Standard recombination operators can be used because exchanging parts of distance matrices will always result in distance matrices that lead to valid trees. The disadvantage of this tree topology representation is its redundancy. Many different distance matrices exist resulting in the same topology. Another disadvantage of the topology representation by means of a distance matrix is the large number of parameters (matrix elements) that have to be optimized. When there are 25 objects, 300 parameters have to be optimized.

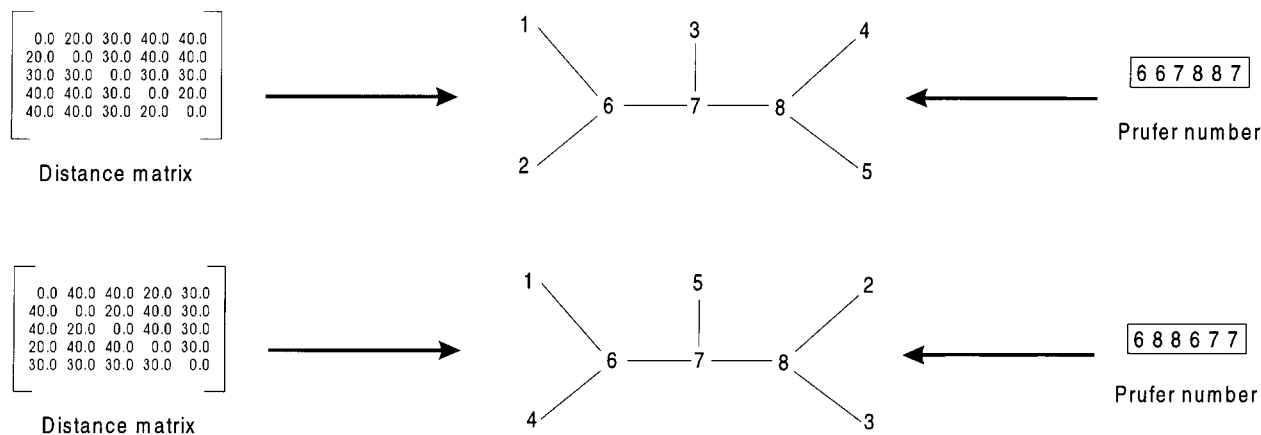


FIGURE 1. Two different representations used to optimize tree topologies. On the left the distance matrix and on the right the Prüfer number representation are depicted. After alteration of the distance matrix or Prüfer number the tree topology is changed.

TOPOLOGY REPRESENTATION II: PRÜFER NUMBER

By using the Prüfer number representation of a tree,¹¹ the number of parameters put on a genetic algorithm string is greatly reduced. The Prüfer number is a $2(n - 2)$ "digit" number where the digits are numbers defined by the following algorithm (n is the number of leaves):

1. Label all leaves and nodes.
2. Let i be the lowest numbered leaf and j the node that is connected to i , then j is the rightmost digit of the Prüfer number.
3. Remove leaf i from the tree and, if needed, update the leaves.
4. If two leaves remain, the Prüfer number is complete; else return to step 2.

An algorithm describing this routine the other way around, from Prüfer number to topology, is also available. Figure 1 shows two bifurcating trees with their corresponding Prüfer numbers. Two important remarks concerning Prüfer numbers have to be made. First, the assigned digit values of the Prüfer number (the content of the Prüfer number) remain the same. Each node label appears two times in the Prüfer number. Second, the topology of the tree is defined by the sequence of the digits in the Prüfer number. A genetic algorithm can be used to optimize the tree topology by optimizing the sequence of the digits in the Prüfer number. To solve this sequential problem, instead of a numerical genetic algorithm a sequential genetic algo-

rithm should be applied using special sequential recombination and mutation operators.^{12, 13}

QUALITY CRITERIA

In ref. 5 four quality criteria are described that can be used to obtain additional information concerning the performance of numerical genetic algorithm optimizations. Here the quality criteria are applied so that the distance matrix representation, which is a numerical approach, can be compared with the Prüfer number representation, a sequential genetic algorithm approach.

The four quality criteria can be summarized as follows:

- *Coverage of the relevant search space.* Many optimization problems exist where, apart from the global optimum, as many as possible near-optimal solutions should be found. This is also the case for phylogenetic clustering problems where it is preferable to have a set of high-quality topologies because the data used are incomplete or may contain errors. By checking the coverage of the relevant search space, insight is gained into how many important optima are found.
- *Reproducibility of the coverage of the relevant search space.* When other starting populations are used, all above-mentioned optima should be found repeatedly.
- *Coverage of the total search space.* If more different solutions are evaluated, the chance of not finding important optima decreases. By

monitoring the variety of observed solutions, a measure for the coverage of the total search space is obtained. In large solution spaces this criterion will be very small, and its importance will decrease somewhat.

- *Reproducibility of the coverage of the total search space.* By repeating the optimization starting from other points in the search space, the same variety of solutions should be explored. In this way the robustness of the coverage of the total search space is tested.

How the above-mentioned criteria are calculated is described in the next section.

Experimental

QUALITY CRITERIA

In order to apply the quality criteria, they have to be adapted such that they can be used in the numerical and sequential genetic algorithm approaches. To obtain the first and second criteria, coverage of the relevant search space, and reproducibility of this coverage, a greedy clustering method as described in ref. 5 is applied to all solutions (strings) from the last population. As a result all similar tree topologies will be put together in one cluster. After counting the number of clusters, an indication of the number of different near-optimal solutions is obtained. This method starts with clustering the strings within a certain distance from the string with the lowest error. This is repeated for the remaining strings until all strings are clustered. The distance measure used to cluster the strings is not based on the parameters put on the strings but on another distance measure independent of the representation used. By counting the number of nodes between the n tips of a topology, a vector consisting of $(n(n-1))/2$ scalars is obtained. The distance between two topologies is obtained by subtracting the two accompanying vectors. Because only the tree topology is optimized by the genetic algorithm, no distance measure is used where also branch length information is incorporated. Of course other tree to tree distance measures can be used. It is important to stress that clustering is performed by using the topologies of the trees, not on basis of the error values. After some initial experiments, a threshold distance value of 5 appeared to be suitable because larger threshold values led to clusters that contain topologies that differ a lot from each other. *The*

first quality criterion is obtained by counting the number of clusters (the number of near-optimal solutions). To improve the reproducibility, in ref. 5 a cluster is defined as containing at least three elements. Because the second quality criterion deals with the reproducibility of the first criterion, here all clusters are counted irrespective of the number of elements of which they consist. The number of clusters containing strings of different runs forms *the second quality criterion*. A measure for *the coverage of the whole search space* is acquired by counting the total number of different tree topologies examined during the whole optimization process. Finally, to gain insight in *the reproducibility of the coverage of the whole search space*, principal component analysis (PCA)¹⁴ is applied on the vector notation of the tree topologies (the number of nodes between the tips) as used during the calculation of distances between tree topologies. With this vector notation the topologies derived from a tree with 10 tips can be plotted in a 45-dimensional space. By applying PCA the number of dimensions can be reduced without losing essential information. PCA constructs new latent parameters called principal components (PCs), from linear combinations of the old parameters. The first latent parameter is chosen in such a way that it explains as much variance of the original data as possible. Each next component is orthogonal to all previous PCs and maximizes the remaining variance. Normally, the first couple of PCs are used to reflect the relevant part of the data. The residuals describe the part of the data that is not explained by the PCs and can be used as a measure of how well the PCs describe the data. On each distance data set five genetic algorithm runs are performed using different starting populations. To get insight into the robustness of the coverage of the whole search space on each run, PCA is applied and the intrarun residuals are calculated. After projection of the topologies of the other runs in the same PCA space, the ratio of the mean interrune residuals versus the mean intrarun residuals is a measure for the robustness of the optimization. If this ratio approximates one, all runs examine the same part of the search space during the optimization.

DATA

Two sets of simulated data are used to examine both genetic algorithm representations. The simulated distance data are calculated using the two topologically extreme model trees depicted in Figure 2 using the same conditions as in ref. 15

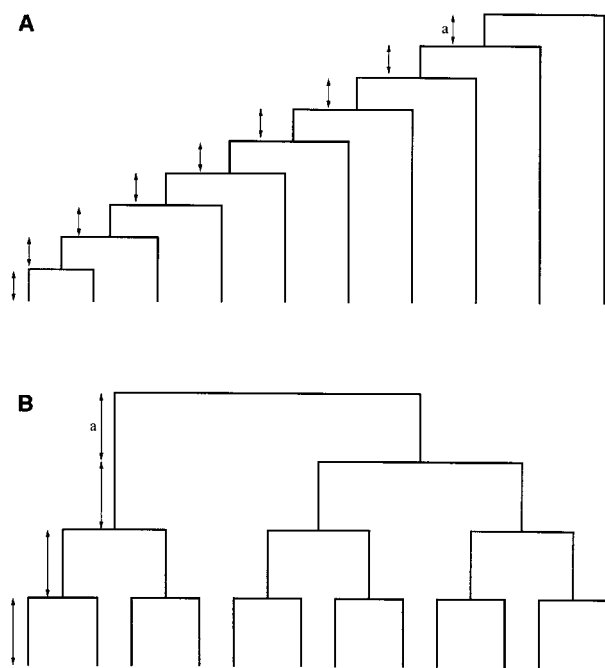


FIGURE 2. Two model trees used to create simulated distance data. In addition to being topologically extreme, both trees have a different number of tips. Model tree A has 10 tips and model tree B has 12 tips. A value of 2 is used for a , the theoretical branch length.

(nucleotide substitution occurs at random with an equal probability). After an ancestor consisting of 300 nucleotides is randomly generated, its descendants are produced by getting the number of mutations from a Poisson distribution with the mean set equal to the theoretical branch length. In this way, five different distance matrices are obtained for each model topology.

SETTINGS AND PROGRAMS

The settings displayed in Table I are used for both genetic algorithm approaches (see refs. 12 and 13 for further explanation). The only settings that are different for both approaches are the crossover and mutation operators used. Furthermore, the manner in which an initial population is created differs. Each optimization is repeated 5 times using another initial population.

The clustering algorithm used in the distance matrix representation is the neighbor-joining algorithm.¹⁶ For a certain tree topology the node to node lengths are determined according to an algorithm developed by Fitch and Margoliash.¹⁷ Once the topology and the branch lengths of the tree are fixed, a new distance matrix can be deduced. By

TABLE I.
General Settings of Genetic Algorithm.

Population size = 30
Max no. generations = 100
Crossover probability = 0.85
Mutation probability = 0.1
Selection type = rank threshold

Settings	Numerical GA	Sequential GA
Crossover type	Uniform crossover	Partially matched crossover
Mutation type	Range mutation	Reorder mutation

comparing this distance matrix with the original distance matrix obtained from pairwise comparisons of DNA or protein sequences, a quality measure can be assigned to the tree. As a criterion for the quality of the reconstruction of the phylogenetic tree, the root mean squared (RMS) relative deviation between the reconstructed (D^r) and original (D) distances is used:

$$\text{RMS} = 100 \left[\frac{1}{\frac{n(n-1)}{2}} \sum_i \left(\frac{D_i^r - D_i}{D_i} \right)^2 \right]^{1/2}. \quad (2)$$

The summation extends over all $n(n-1)/2$ distances between the n leaves.

All the algorithms used to calculate the RMS value of a tree topology and the genetic algorithm software package, PGAPACK, are written in ANSI C. (PGAPACK can be obtained by anonymous ftp from <ftp://ftp.mcs.ano.gov/pub/pgapack/>.) The quality criteria were calculated using Matlab for Unix Workstations, version 4.2c, by MathWorks, Inc. The genetic algorithm and Matlab calculations were performed on a Sparc workstation (SUN). One run of the genetic algorithm required ~ 2 min.

Results and Discussion

Figure 3 depicts two error plots typically observed during the optimization of model tree B. These kinds of plots are often used to assess the quality of an optimization. In the upper part of the figure the best error of five replicate runs of the distance matrix representation is monitored. Eventually all runs result in the optimal topology. The Prüfer number representation (lower part of the figure) seems to perform slightly worse; one run

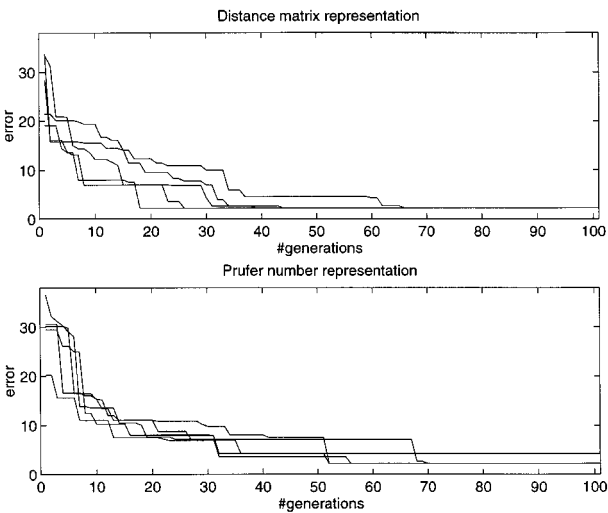


FIGURE 3. Two error plots typically found during the phylogenetic tree optimization of a data set of model tree B. The top part shows the decrease of the best error during the optimization of the five independent runs for the distance matrix representation. The bottom part shows the same information for the Prüfer number representation.

converges prematurely and results in a less-optimal topology. For the first model tree both representations give similar results. In all runs the topology resulting in the lowest error was found, so those error plots are not shown here.

Palmer and Kershenbaum concluded that the Prüfer number representation would be less effective because of missing locality (i.e., small changes in the representation do not result in small changes in the topology).¹⁸ These results confirm the outcome of the experiments performed by Palmer and Kershenbaum, so a preference for the distance matrix representation seems justified. All error plots of both representations show behavior typically found during optimization with genetic algorithms.

In Figure 4 the topologies of the trees with the lowest error are shown for each of the five different data sets (model tree A). None of the found lowest error trees has the same topology as the used model tree. This illustrates the importance of obtaining a set of good quality solutions. In case of the second model tree all found topologies equal the model tree topology, so these results are not shown here.

As mentioned in the Experimental section, the first and second quality criteria (the coverage and the reproducibility of the coverage of the relevant

search space) are obtained after applying a greedy clustering technique on the last population of the different runs. Figure 5 gives the results of this clustering for one data set of model tree A for both the numerical and the sequential genetic algorithm approach. On the *x* axis of the figure the cluster centers are depicted in such a way that the first cluster center has the lowest error and the last cluster center has the highest error. If a cluster center is absent for a certain representation, the genetic algorithm does not evaluate topologies with corresponding errors. The *y* axis returns the number of objects situated in the corresponding cluster. Only the first 20 centers are represented because we are mainly interested in topologies with low errors. It is striking to see that in the distance matrix representation the first cluster dominates the whole plot. For the Prüfer number representation this is not the case: it also samples tree topologies with errors close to the lowest error. For the data sets of model tree B similar results are obtained.

In Figure 6 the error values of the cluster centers are plotted for both model trees (error values are obtained from both representations). For model tree A, the error values of the nearly optimal trees differ very little from the optimal tree. In case of model tree B the error differences between the different tree topologies are much larger. This is also the reason why the found optimal tree topologies for model tree A do not match the model topology.

Figure 5 and the clustering plots of the other simulated data sets clearly demonstrate that the distance representation examines far fewer nearly optimal tree topologies while also finding the optimal topology than the Prüfer number representation does. This can also be concluded after the coverage of the relevant search space is obtained by counting the number of clusters (the number of “good” solutions). For the given plots this results in 12 and 19 clusters for the distance and Prüfer representation for model tree A, respectively. In Table II the results for all simulated data sets are summarized. For all sets of data the Prüfer representation finds more clusters and accordingly finds more nearly optimal tree topologies than the distance matrix representation.

The robustness of the coverage of the relevant search space is obtained by counting the number of mixed clusters (clusters containing objects of different runs). The results are shown in Table III. Again, in general, the Prüfer number representation performs better.

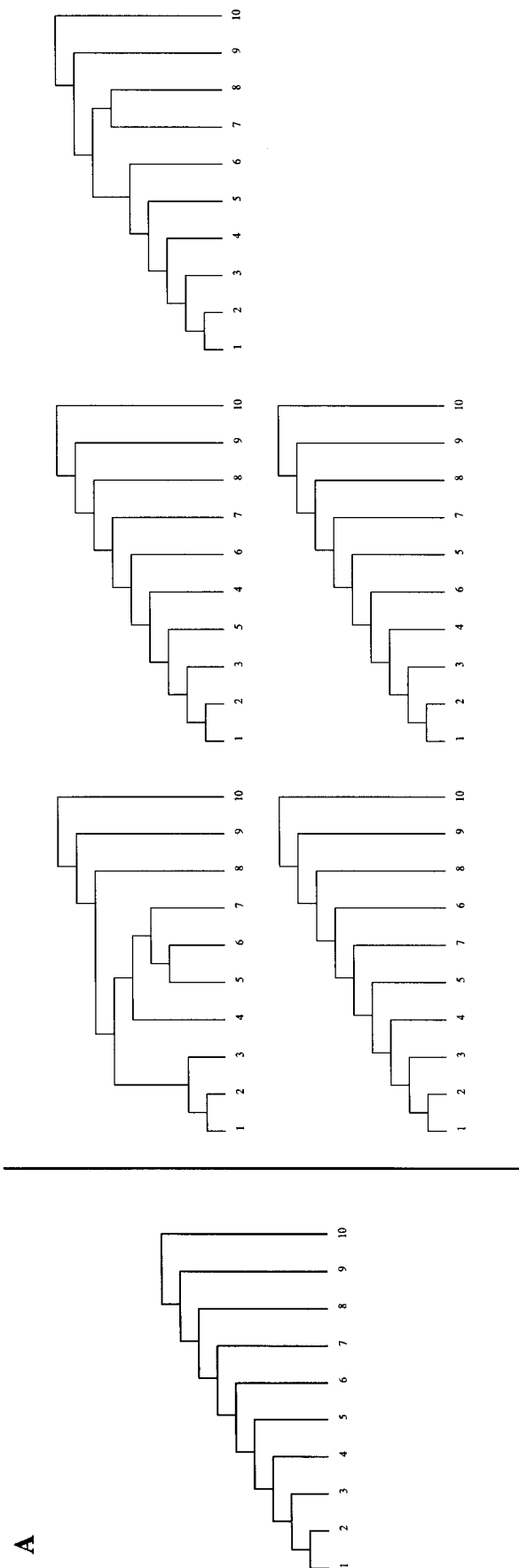


FIGURE 4. Five topologies of the trees with the lowest error found by the genetic algorithm. On the left-hand side of the figure the model tree (model tree A) is depicted.

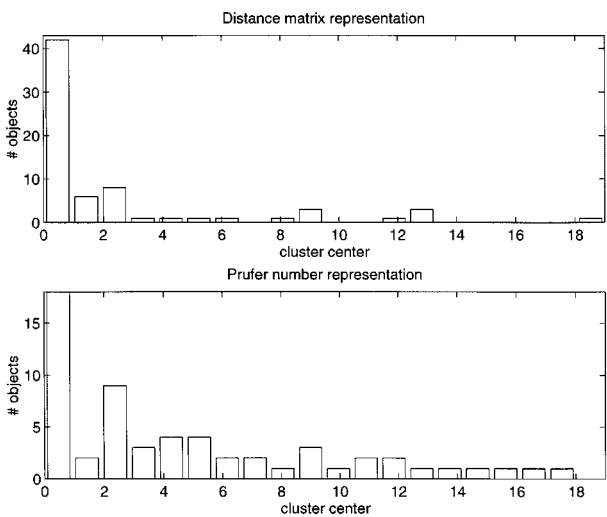


FIGURE 5. Results after clustering a data set containing all topologies evaluated by the five independent runs during the last generation. On the x axis of the figure the cluster centers (the near-optimal topologies) are depicted. The first cluster has the lowest error value while the last cluster has the highest error value. The y axis returns the number of objects situated in the corresponding cluster. On top the results of the distance matrix representation are given. The bottom part shows the results for the Prüfer number representation.

In addition to the coverage of the relevant search space, the coverage of the whole search space is considered. According to eq. (1), the total number of different tree topologies is approximately 2 million and 646 million for bifurcating trees with 10 (model tree A) and 12 (model tree B) tips, respec-

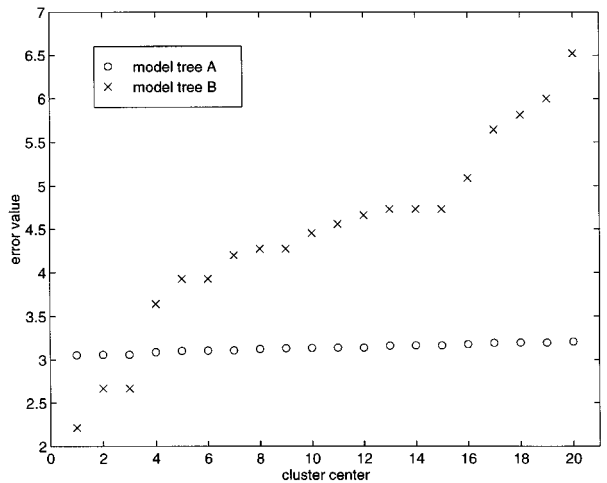


FIGURE 6. Error values of the cluster centers for model tree (o) A and (+) B.

TABLE II.
First Quality Criterion: Coverage of Relevant Search Space.

Representation	Data Set									
	Model Tree A					Model Tree B				
	1	2	3	4	5	1	2	3	4	5
Distance	12	14	16	11	13	8	9	11	13	10
Prüfer	19	18	17	17	18	17	18	17	15	19

The number of clusters (nearly good solutions) found after applying greedy clustering. A minimal cluster size of 1 was used.

tively. The more topologies are examined during the optimization, the larger the probability that the optimal tree will be found. The coverage of the whole search space is expressed by counting the number of different topologies encountered during the whole optimization. Table IV gives the number of different tree topologies averaged over the five replicate runs. The maximum number of different tree topologies that can be tested during each genetic algorithm optimization with a population size of 30 and a maximum number of generations of 100 (see Table I) equals 3000. Both representations evaluate considerably fewer different tree topologies before reaching the optimal tree. In spite of its redundant character, the distance matrix representation examines more different topologies than the Prüfer number representation. An explanation for this observation cannot be given.

In order to get a measure for the robustness of the coverage of the whole search space, PCA was applied on all strings evaluated during the optimization. The ratio of the mean of the interr run residuals and the intrarun residuals is given in Table V.

TABLE III.
Second Quality Criterion: Reproducibility of Coverage of Relevant Search Space.

Representation	Data Set									
	Model Tree A					Model Tree B				
	1	2	3	4	5	1	2	3	4	5
Distance	4	5	7	7	7	5	2	5	3	2
Prüfer	8	9	7	10	7	9	4	5	5	4

The number of mixed clusters found after applying greedy clustering.

TABLE IV.
Third Quality Criterion: Coverage of Total Search Space.

Representation	Data Set									
	Model Tree A					Model Tree B				
	1	2	3	4	5	1	2	3	4	5
Distance	1321	1378	1356	1390	1233	1811	1800	1806	1711	1906
Prüfer	1111	1120	1055	1118	1107	1467	1473	1410	1393	1453

The number of different topologies examined during the whole optimization.

On basis of these findings no clear preference for one of the representations can be given. In case of model tree A the Prüfer number representation performs better while for model tree B the distance matrix representation gives better results. The differences are small for both cases. However, if the score plots are considered both representations behave dissimilarly. In Figure 7 for data set 1 of model tree A, the best and worst topologies are depicted using the defined PC1 and PC2 axes. Figure 7A visualizes the topologies of the distance matrix representation while Figure 7B shows the topologies of the Prüfer number representation. Although in Figure 7A no clear difference between the good and bad topologies can be detected, in Figure 7B the good and bad topologies are more or less clustered or separated. The opposite occurs (results not shown) for model tree B; in the PCA space constructed with topologies of the distance matrix representation good and bad solutions are ordered, while in the PCA space constructed with topologies of the Prüfer number representation this is not the case. The differences between the ratio values in Table V may be small, but if the corresponding score plots are considered it can be concluded that these differences still result in completely different PCA results. Nevertheless, on the basis of the last quality criterion, no clear preference for one of the representations can be

given. Further research concerning these results is underway.

Conclusion

Experiments were performed to obtain a measure of performance for the two different tree topology representations. In addition to monitoring the fitness of the best string of each population, several other quality criteria were applied. The experiments show that on the basis of these criteria the suitability of the used representations can be estimated.

On the basis of the error plots a preference for the distance matrix representation is justified. However, the first and second criteria clearly emphasize that the Prüfer number representation samples far more near-optimal topologies while also finding the global-optimal topology. So if one is interested in obtaining a set of good quality trees, it is advisable to use the Prüfer number representation. If only the optimal tree topology is wanted, the third quality criterion suggests using the distance matrix representation. Further research is being done where on basis of information obtained from the quality criteria, the representations are adjusted so that the efficiency is improved.

TABLE V.
Fourth Quality Criterion: Reproducibility of Coverage of Total Search Space.

Representation	Data Set									
	Model Tree A					Model Tree B				
	1	2	3	4	5	1	2	3	4	5
Distance	1.2153	1.2779	1.1989	1.2985	1.1935	1.2329	1.1833	1.1716	1.1157	1.2680
Prüfer	1.1167	1.1719	1.1510	1.1603	1.0972	1.2748	1.2530	1.2577	1.2349	1.2753

Ratio of between-run and within-run residuals. Six principal components are used.

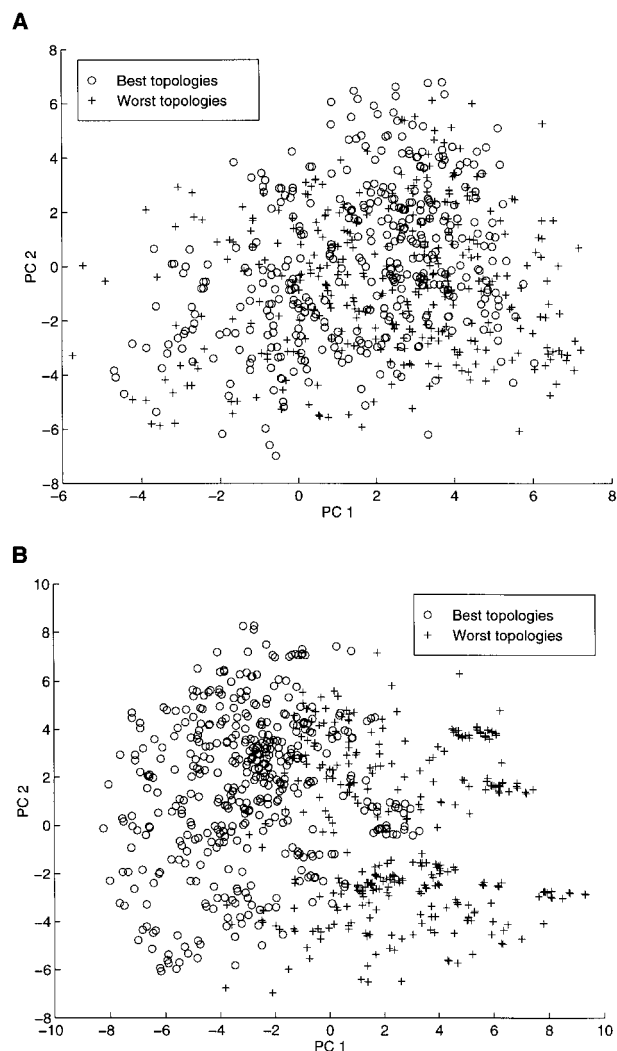


FIGURE 7. Score plots for (A) the distance matrix representation and (B) the Prüfer number representation; (○) the best topologies and (+) the worst topologies are represented.

References

1. Swofford, D. L.; Olsen, G. J. In *Molecular Systematics*; Hillis, D. M.; Moritz, C., Eds.; Sinauer Associates: Sunderland, CA, 1990; pp 411–501.
2. Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*; Addison-Wesley: Reading, MA, 1989.
3. Holland, J. H. *Adaptation in Natural and Artificial Systems*; MIT Press: Cambridge, MA, 1992.
4. Massart, D. L.; Vandeginste, B. G. M.; Buydens, L. M. C.; de Jong, S.; Lewi, P. J.; Smeyers-Verbeke, J. In *Handbook of Chemometrics and Qualimetrics: Part A*; Elsevier: Amsterdam, 1997; pp 805–845.
5. Wehrens, R.; Pretsch, E.; Buydens, L. M. C. *J Chem Inf Comput Sci* 1998, 38, 151–157.
6. de Weijer, A. P.; Lucasius, C. B.; Buydens, L.; Kateman, G.; Heuvel, H. M.; Mannee, H. *Anal Chem* 1994, 66, 23–31.
7. Dane, A. D.; van Sprang, H. A.; Buydens, L. M. C. *Anal Chem*, to appear.
8. Matsuda, H. In *Proceedings of the Genome Informatics Workshop 1995*; Hagiya, M.; Suyama, A.; Takagi, T.; Nakai, K.; Miyano, S.; and Yokomori, T., Eds; Universal Academy Press: Tokyo, 1995; pp 19–28.
9. Lewis, P. O. *Mol Biol Evolution* 1998, 15, 277–283.
10. Reijmers, T. H.; Wehrens, R.; Daeyaert, F. D.; Lewi, P. J.; Buydens, L. M. C. *BioSystems* 1998, 49, 31–43.
11. Moon, J. W. In *A Seminar on Graph Theory*; F. Harary, Ed.; Holt, Rinehart, & Winston: New York, 1967; pp 70–78.
12. Lucasius, C. B.; Kateman, G. *Comput Chem* 1994, 18, 127–136.
13. Lucasius, C. B.; Kateman, G. *Comput Chem* 1994, 18, 137–156.
14. Jackson, J. E. *A User's Guide to Principal Components*; Wiley: New York, 1991.
15. Tateno, Y.; Nei, M.; Tajima, F. *J Mol Evolution* 1982, 18, 387–404.
16. Saitou, N.; Nei, M. *Mol Biol Evolution* 1987, 4, 406–425.
17. Fitch, W. M.; Margoliash, E. *Science* 1967, 155, 279–284.
18. Palmer, C. C.; Kershbaum, A. In *International Conference on Evolutionary Computation '94*; IEEE Press: New York, 1994; pp 439–444.